PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-338824
(43)Date of publication of application : 10.12.1999

(51)Int.Cl. G06F 15/00
G06F 13/00

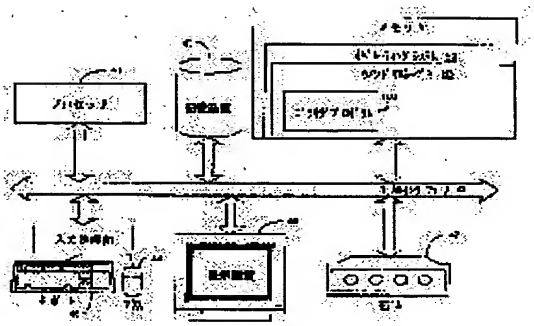
(21)Application number : 11-102757
(22)Date of filing : 09.04.1999

(71)Applicant : HEWLETT PACKARD CO <HP>
(72)Inventor : STEELE DOUGLAS W
GOIN TODD M
BRYANT CRAIG W

(30)Priority
Priority number : 98 60875 Priority date : 15.04.1998 Priority country : US

(54) METHOD FOR PROTECTING WEB RESOURCE

(57)Abstract:
PROBLEM TO BE SOLVED: To achieve flexible high-grade security in the case of accessing a web resource on a by a cliant system.
SOLUTION: In the client system, a memory 51 is provided with an operating system 52 and a window manager 53. A token is generated by a client browser 100 inside the memory 51 of a computer and transferred to the security server together with a service request. Besides, these service request and token are supplied to a server application, the verification of the token is requested, the received token is transferred to the security server and tokens from the client browser 100 and the server application are compared. In the case of coincidence in this comparison, a coincidence notice is generated but when these tokens are not coincident, a non-coincidence notice is generated and transferred to the server application. When the coincidence notice is received, the requested service is provided.



LEGAL STATUS

[Date of request for examination] 19.12.2001
[Date of sending the examiner's decision of rejection]
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]
[Date of final disposal for application]
[Patent number]
[Date of registration]
[Number of appeal against examiner's decision of rejection]
[Date of requesting appeal against examiner's decision of rejection]

(19)日本国特許庁（ＪＰ）　(12)**公開特許公報（Ａ）**　(11)特許出願公開番号

特開平11－338824

(43)公開日　平成11年(1999)12月10日

(51)Int.Cl.[6]		識別記号	FI		
Ｇ０６Ｆ	15/00	３３０	Ｇ０６Ｆ	15/00	３３０Ａ
	13/00	３５４		13/00	３５４Ｚ

審査請求　未請求　請求項の数１　ＯＬ　（全 12 頁）

(21)出願番号　特願平11－102757

(22)出願日　平成11年(1999)４月９日

(31)優先権主張番号　０９／０６０－８７５

(32)優先日　1998年４月15日

(33)優先権主張国　米国（ＵＳ）

(71)出願人　398038580
ヒューレット・パッカード・カンパニー
ＨＥＷＬＥＴＴ－ＰＡＣＫＡＲＤ　ＣＯＭＰＡＮＹ
アメリカ合衆国カリフォルニア州パロアルト　ハノーバー・ストリート　3000

(72)発明者　ダグラス・ダブリュー・スティーレ
アメリカ合衆国　コロラド，フォートコリンス，マックマリー　5224

(74)代理人　弁理士　萩野　平　（外４名）

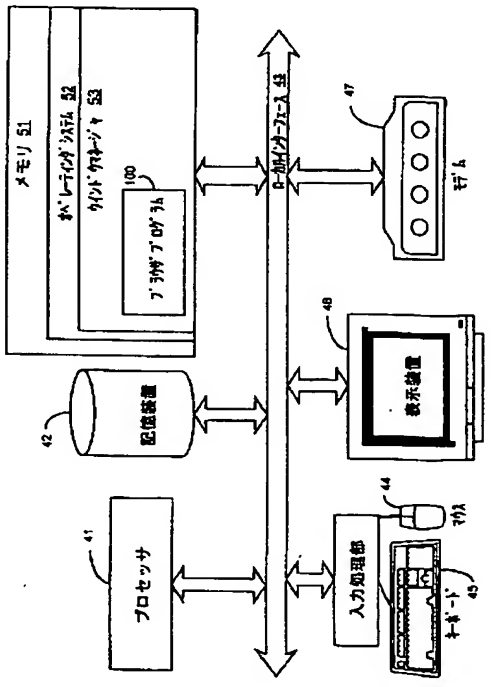
最終頁に続く

(54)【発明の名称】　ウェブリソースの保護方法

(57)【要約】

【課題】　クライアントシステムによって、サーバ上のウェブリソースにアクセスする際に、融通性のある、かつ高度なセキュリティ実施を可能にする。

【解決手段】　クライアントシステムにおいて、メモリ５１は、オペレーティングシステム５２とウインドウマネージャ５３とを備えている。コンピュータのメモリ５１内のクライアントブラウザ１００でトークンを生成し、セキュリティサーバに対してサービス要求とともに転送する。また、このサービス要求及びトークンをサーバアプリケーションに供給してトークンの検証を要求し、受信したトークンをセキュリティサーバに転送して、クライアントブラウザ１００及びサーバアプリケーションからのトークンを比較する。この比較で一致した際に一致通知を生成し、一致しない場合に不一致通知を生成してサーバアプリケーションに転送する。一致通知を受信した際に要求されたサービスを提供する。



1

【特許請求の範囲】
【請求項1】　通信ネットワークシステムにおけるウェブリソースの保護方法であって、
クライアントブラウザ(１００)に対してトークンを生成するステップと、
セキュリティサーバ(１４０)に対して前記トークンを転送するステップと、
サービス要求及び前記トークンを転送するステップと、
前記クライアントブラウザ(１００)からの前記サービス要求及び前記トークンを受信するアプリケーションサーバ(１６０)を提供するステップと、
前記アプリケーションサーバ(１６０)によって前記トークンの検証を要求するステップと、
前記アプリケーションサーバ(１６０)が受信した前記トークンを前記セキュリティサーバ(１４０)に転送するステップと、
前記クライアントブラウザ(１００)から受信した前記トークンと前記アプリケーションサーバ(１６０)から受信した前記トークンとを比較するステップと、
を有することを特徴とするウェブリソースの保護方法。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】本発明は、コンピュータ及びソフトウエアに関し、特に、クライアントブラウザ(client browser)によってサーバ上のウェブリソース(Web resource)にアクセスする際に伴うセキュリティ(保護)を実施するウェブリソースの保護方法に関する。
【０００２】
【従来の技術】従来、インターネットは、通信プロトコルのＴＣＰ／ＩＰ(Transmission Control Protocol/Internet Protocol)を適用して相互通信を行う通信ネットワーク及びゲートウエー(gateway)を接続した世界的なコンピュータ・ネットワーク集合体である。インターネットは、その中心として、データ及びメッセージを伝送する膨大な商用、政府、教育用及びその他のコンピュータシステムからなる主ノード又はホストコンピュータ間を高速データ通信回線で接続している。
【０００３】WWW(World Wide Web)は、世界中のＨＴＴＰ(Hypertext Transfer Protocol)を用いたサーバ(以後、ＨＴＴＰサーバと記載する)にインターリンク(interlink)されたハイパーテキストの文書の全体を参照する。WWW上の文書は、ページ又はウェブ(Ｗｅｂ)ページと呼称され、インターネット上のロケーション(location)を示すＵＲＬ(Uniform Resource Locator)で識別されるＨＴＭＬ(HyperText Markup Language)によって記載されている。このＵＲＬは、特定のマシン及びパス名を指定し、このＵＲＬによって、ＨＴＴＰに基づいたファイルへのアクセス、及び、ファイルのノード間からエンドユーザへの伝送を可

2

能にしている。ウェブサイト(Web site)は、これらの文書と、WWW上のＨＴＴＰサーバによって提供される関連ファイル、スクリプト(script)、サブプロシージャ(subprocedure)及びデータベースとの関連グループである。
【０００４】ユーザは、ウェブサイトにアクセスするために、ブラウザプログラム(browser program)とインターネット接続とを必要とする。ブラウザプログラムは、「ウェブブラウザ(web browser)」とも呼ばれ、ユーザが、インターネットをナビゲート(navigate)し、WWW、他のネットワーク又はユーザのコンピュータ上でＨＴＭＬ文書を閲覧できるようにするためのクライアントアプリケーション(client application)である。また、ブラウザプログラムによってユーザは、ＨＴＭＬ文書内に組込まれている「タグ(tag)」と呼ばれるコードを指定した処理が実行できる。このタグは、文書内の特定の語及び画像をＵＲＬと関連付けることによって、ユーザが、キーを押下したとき、又は、マウスをクリックしたときに、世界中のどこかにある他のファイルにアクセスできるようにするためのものである。
【０００５】これらのファイルには、Java applet、perl application、他のスクリプト言語(scripted language)、active X-control、又は、ユーザがリンク上でクリックすることによって起動すると実行する他の小容量の内蔵型ソフトウエアプログラムと同様に、テキスト(あらゆるフォント及びスタイルのもの)と、グラフィック画像と、映像ファイルと、サウンドとが含まれる。スクリプトは、クライアントユーザ(client user)による要求に応じてＨＴＴＰサーバが実行するアプリケーションである。これらのスクリプトは、単一ジョブを行うＨＴＴＰデーモン(HTTP daemon)によって呼出され、その後終了する(exit)。
【０００６】スクリプトの一つのタイプとして、共通ゲートウエーインタフェース(common gateway interface,ＣＧＩ)スクリプトが周知である。ＣＧＩスクリプトは、一般的にユーザがリンク又は画像などのウェブページ内の要素をクリックした際に、呼び出される。ＣＧＩスクリプトを使用することによって、ウェブページ内での対話性が得られる。このＣＧＩスクリプトは、Ｃ、Ｃ＋＋、Ｐｅｒｌのような多くの言語で記載できる。また、ＣＧＩ－ＢＩＮとは、ＨＴＴＰサーバが実行することが出来るＣＧＩスクリプトのアプリケーションのライブラリ(library)である。
【０００７】WWW上のＨＴＴＰサーバによって提供されるこれらの文書及び関連ファイル、スクリプト、サブプロシージャ及びデータベースに対してアクセスする場合、セキュリティが重要な問題となる。すなわち、許可されたクライアントシステムからの許可されたユーザのみに対して、ＨＴＴＰサーバのアプリケーションへのアクセスを許可するように、その検証をどのようにする

か、また、悪意のある目的でアクセスを不正使用できないようにする保証をどのようにするかの課題がある。

【０００８】この課題に対して、現在使用されている方法に、「Ｃｏｏｋｉｅ」を使用したものがある。Ｃｏｏｋｉｅは、サーバが、クライアントからの要求に応じてクライアントに返送するデータのブロック（以後、データブロックと記載する）である。このデータブロックは、クライアントシステムに格納されている。クライアントは、同一のウェブサイトに戻るときに、Ｃｏｏｋｉｅのコピーをサーバに送信し、この送信によってサーバがクライアントを識別する。Ｃｏｏｋｉｅは、ユーザを識別したり、サーバに対して要求されたウェブページのカスタマイズされたバージョンを送信するよう命令したり、ユーザにアカウント情報を提示したりする場合の管理目的に使用される。ほとんどのシステムでは、Ｃｏｏｋｉｅプログラムはユーザのログオン中に稼動する。

【０００９】このようなウェブリソースにアクセスする際にセキュリティを提供する従来の方法には、以下のようなセキュリティに関する欠点がある。本発明がいかにこれらの課題に取り組んで解決しているかについては、この後で説明する。

【００１０】

【発明が解決しようとする課題】従来の解決法の問題は、ホストアドレス及びユーザ名（すなわち、ユーザのログオン情報）が、非常に「スプーフィング（spoofing）」を受け易い普通テキストで送信されていることである。それゆえ、精通したハッカーは、他のマシン又は他のユーザからのように見せかけたパケット伝送によって無許可でサーバにアクセスすることができる。

【００１１】また、複数レベルのユーザのセキュリティが試みられている場合、他の問題が発生する。すなわち、Ｃｏｏｋｉｅを用いた方法では、セキュリティの単一レベルにのみ対応が可能である。更に、Ｃｏｏｋｉｅを使用する場合、ユーザアプリケーションはセキュリティシステムに統合することが出来ない。現在、Ｃｏｏｋｉｅはクライアントのブラウザプログラムの一部であり、ユーザアプリケーションからは独立している。

【００１２】従来技術における他の問題は、セキュリティに対する検証（authentication）が的確に出来ないことである。これは、サーバが、証明（proof）無しに、伝送中に識別されるユーザ名及びホスト名を受け入れるからである。更に、各コマンドのトランザクション（transaction）が独立しているため、いかなる状態も維持されないという問題がある。これによって、これらの方法が、「リプレイアタック（replay attack）」（ハッカーが正当なネットワークパケットを獲得し、いくつかの詳細（ユーザ名又は実行コマンド等）を変更して再度送信する）を受け易くなる。

【００１３】このように、上記従来例では、今日までネットワークシステムは、インターネット又は他の種類の

ネットワーク上のウェブ文書（又ウェブリソースは）に対して、融通性（flexible）を備え、かつ、高度なセキュリティの実施を提供できないという欠点があった。

【００１４】本発明は、このような従来の技術における課題を解決するものであり、クライアントブラウザによって、サーバ上のウェブリソースにアクセスする際に、融通性のある、かつ高度なセキュリティの実施が可能になるウェブリソースの保護方法の提供を目的とする。

【００１５】本発明のある目的、利点及び新規な特徴について、一部は以下で説明するが、この一部は以下の内容から、その調べによって当業者で明らかになるものである。あるいは、本発明を実施することによって知ることが出来るものである。本発明の目的及び利点は、特に特許請求の範囲で示す手段及び組合せによって、実現できるものである。

【００１６】

【課題を解決するための手段】上記課題を達成するために、本発明の一実施形態によれば、クライアントシステムのクライアントユーザインタフェース（ブラウザプログラム）がトークン（token）を生成する。このトークンは、セキュリティサーバ（security sever）へ送信されて、クライアントユーザのサービス要求に対する第三者による妥当性の検証（third party validation）を行う。そして、クライアントユーザインタフェースは、サービスのためにサーバアプリケーション（sever application）を呼び出し、サービスのための呼び出しの引数（argument）としてトークンをサーバアプリケーションに対して送信する。

【００１７】サーバアプリケーションは、クライアントユーザインタフェースからのサービス要求を受信し、その後クライアントユーザについてそれ自身のログイン検証プログラム（login authorization program）を実行する。ここで検証の結果がＯＫである、すなわちログインが正当である場合、サーバアプリケーションは、要求されたサービスについて必要なＣＧＩ－ＢＩＮアプリケーションプログラム（以後、ＣＧＩ－ＢＩＮプログラムと記載する）に対する呼び出しを実行する。

【００１８】要求されたサービスのために呼出されたＣＧＩ－ＢＩＮプログラムは、プログラム名等のサービス識別子（service indicator）及びクライアントユーザインタフェースが生成したトークンが含まれる引数を受信する。要求されたプログラムは、セキュリティサーバへの接続を確立し、その後、引数として受信したトークンを、検証のためにセキュリティサーバへ送信する。

【００１９】セキュリティサーバは、要求されたプログラムから検証するトークンを受信し、要求されたプログラムを実行して受信したトークンを、クライアントユーザインタフェースを通じて受信したトークンを用いて検証する。セキュリティサーバは、この二つのトークンが一致した場合に、要求されたプログラムに対し、そのト

5

ークンが照合(verify) したとの指示を返送する。セキュリティサーバは、要求されたプログラムについてトークンを検証した後に、直ちにクライアントユーザインタフェースからトークンを受信する待ち状態に戻る。

【００２０】CGI －BINプログラムは、要求されたプログラムを実行し、その終了前にサーバアプリケーションに対して出力を送信する。サーバアプリケーションは、要求されたプログラムの出力を受信し、クライアントユーザに表示するためにクライアントユーザインタフェース(ブラウザプログラム) に、そのデータを返送する。この時点で、サーバアプリケーションは、クライアントユーザインタフェースからのサービス要求を待つ状態に戻る。

【００２１】本発明の他の実施形態によれば、ウェブリソースの保護のために、複数レベルのユーザのセキュリティを実施する。

【００２２】本発明の他の実施形態によれば、ウェブリソースの保護を実現する装置及び方法では、ユーザアプリケーションに対し、統合されたセキュリティシステムを提供する。

【００２３】このような本発明のウェブリソースの保護方法は、クライアントブラウザによって、サーバ上のウェブリソースにアクセスする際に、融通性のある、かつ高度なセキュリティの実施が出来るようになる。

【００２４】本明細書に組込まれその一部を形成する添付図面は、本発明のいくつかの様相を説明しており、明細書中の説明とともに、本発明の原理を説明している。

【００２５】

【発明の実施の形態】本発明は、これらの図面に関連して説明するが、その中で開示した実施の形態に限定する意図はない。この反対に、本発明は、特許請求の範囲で定義する本発明の技術思想及び範囲内に含まれる全ての代替物、変形例及び均等物を全て包含することを意図している。

【００２６】次に、本発明のウェブリソースの保護方法の実施の形態を図面を参照して詳細に説明する。

【００２７】図1は、本実施形態に係るインターネットを利用するクライアント／サーバシステムの一構成を示す構成図であり、本発明の融通性と、拡張性と、プラットフォーム(platform) の独立性とを説明している。このようなシステム構成は多くの形態で実現できるが、図1では、ネットワーク１８、例えば、ローカルエリアネットワーク(ＬＡＮ、なお、ＬＡＮに限定されない) に直接に接続されたUNIX又はＰＣ等の複数の異なったワークステーション１２，１６を有している。追加のラップトップ型のワークステーション２１，２２は、同様に遠隔地に配置されており、ダイヤルイン(dial-in) 又は他のネットワーク接続処理手順(通信プロトコル) ２４によってネットワーク１８と通信を行うようになっている。図1では、クライアント装置に該当する各

6

ワークステーション１２，１６，２１，２２が異なったハードウエアプラットフォームで構成しても良いことを強調するために、各ワークステーション１２，１６，２１，２２を一意的に図示した。

【００２８】周知のように、ブラウザアプリケーション(browser application) は、種々のハードウエアのプラットフォームに提供され、容易な使用が可能である。ブラウザは、ブラウザプログラムを用いて、インターネット３２によって情報にアクセスするためのユーティリティ(utility) を有していることで最も一般的に認められている。このブラウザは、ユーザがあらゆるサービス集団と考えることが出来る装置又はプラットフォームであり、ＨＴＴＰを使用して、ウェブサーバ３１又はネットワークサーバ２６から情報を検索し、ＨＴＭＬコードを解釈し、フォーマットして、その翻訳結果をワークステーションの表示装置で表示する。

【００２９】追加のワークステーション３３，３４は、同様に配置されており、ローカルサーバ及びインターネット３２上のウェブページにアクセスするために、ウェブサーバ３１と通信を行うようになっている。ワークステーション３３，３４は、ＬＡＮ３５上のウェブサーバ３１との間での通信を行う。ネットワーク１８及びＬＡＮ３５は、例えば、１０ＢＡＳＥ２，１０ＢＡＳ５，１０ＢＳＡＦ，１０ＢＡＳＴ，ＢＡＳＥ　ＢＡＮネットワーク、及び、ＣＯ－ＥＸケーブル等として周知のイーサネット型ネットワークであっても良い。なお、以下、上記ワークステーション１２，１６，２１，２２，３３，３４は、クライアントシステムと記載し、ネットワークサーバ２６及びウェブサーバ３１をともにサーバシステムと記載する。また、ネットワーク１８及びＬＡＮ３５をネットワーク１８，３５と記載する。

【００３０】図2は、実施形態にあってクライアントシステムのコンピュータシステム内のブラウザプログラム１００を示すブロック図であり、ワークステーション１２，１６，２１，２２，３３，３４のクライアントシステムのアーキテクチャを示している。図2に示すように、一般的に、クライアントシステムには、ネットワーク１１上のロケーション(location) へのアクセス用としてクライアントブラウザのブラウザプログラム１００(例えば、ネットスケープ(Netscape) 、インターネットエクスプローラ(Internet Explorer) 、又は他のブラウザプログラム等) のみが用いられている。これらのブラウザプログラム１００は、メモリ５１内に格納され、かつ、通信機能のモデム４７からアクセスして、ネットワーク１１に接続された他のウェブリソースをユーザに移送(トランスポート (transport)) する。ウェブリソースを見つけるために、ユーザは、ＵＲＬによって示されるウェブリソースのネットワーク１１上のロケーションを知らなければならない。これらのＵＲＬは、しばしば暗号のようであり、それらの名前付けの規則に

7

おける非常に複雑な体系及びフォーマットに基づいている。

【００３１】このようなクライアントシステムでは、プロセッサ４１と、記憶装置４２と、オペレーティングシステム５２及びウインドウマネージャ５３を備えたメモリ５１とを使用して、ユーザが所望するウェブリソースを識別し、かつ、アクセスして処理する。プロセッサ４１は、バス（ローカルインターフェース）４３を通じてメモリ５１及び記憶装置４２からデータを受信する。ユーザからの操作指示は、入力装置のマウス４４及びキーボード４５を使用して信号化される。動作入力及び結果出力は、表示装置４６で表示される。

【００３２】図2に示すように、本実施形態では、ウインドウマネージャ５３にブラウザプログラム１００が設けられている。ブラウザプログラム１００は、サーバと対話して、クライアントユーザが要求したデータ得るとともに、機能処理を実行するソフトウエアである。このブラウザプログラム１００の詳細については、以降の図4及び図5を参照して説明する。

【００３３】図3は、実施形態にあってサーバシステムのコンピュータシステム内のサーバアプリケーションプログラム（以後、サーバアプリケーションと記載する）１２０とＣＧＩ－ＢＩＮアプリケーションプログラム１６０（以後、単にＣＧＩ－ＢＩＮプログラムという。）とセキュリティサーバ１４０とを示すブロック図であり、ネットワークサーバ２６及びウェブサーバ３１（図１参照）のサーバシステムでのアーキテクチャを示している。図1に示すサーバシステム２６，３１とクライアントシステム１２，１６，２１，２２，３３，３４との主な違いは、クライアントシステムが、クライアントユーザにインタフェースし、ブラウザプログラム１００によって機能性を要求するのに対し、サーバシステム２６，３１は、サーバアプリケーション１２０と、セキュリティサーバ１４０と、アプリケーションサーバに該当するＣＧＩ－ＢＩＮプログラム１６０とを利用して、クライアントが要求するサービスを提供する点が相違する。

【００３４】この相違する以外では、サーバシステムは、プロセッサ６１と、記憶装置６２と、マウス６４と、キーボード６５と、表示装置６６と、モデム６７との機能が、基本的に前記した図2のクライアントシステムの対応する要素と同一である。本技術分野で周知のように、クライアントシステム１２，１６，２１，２２，３３，３４及びサーバシステム２６，３１は、同様な物理的なマシン上にある。

【００３５】サーバシステムにおける主な相違は、オペレーティングシステム７２及びウインドウマネージャ７３と対話するためのメモリ７１が、ＨＴＴＰデーモン（以後、ＨＴＴＰＤと記載する）によって呼び出されるサーバアプリケーション１２０と、ＣＧＩ－ＢＩＮプログラム１６０と、セキュリティサーバ１４０とを利用して、クライアントが要求するサービスを提供する点である。このサーバアプリケーション１２０と、ＣＧＩ－ＢＩＮプログラム１６０と、セキュリティサーバ１４０との詳細については、以降の図4、図6、図7及び図8を参照して説明する。

8

【００３６】図4は、クライアントユーザインターフェース（ブラウザプログラム）１００と、サーバアプリケーション１２０と、セキュリティサーバ１４０と、ＣＧＩ－ＢＩＮプログラム１６０とのプロセスを示すブロック図である。図4に示すように、クライアントシステム１２，１６，２１，２２，３３，３４は、ブラウザプログラムのクライアントユーザインターフェース１００を利用することによって、サーバシステム３１にサービスを要求することが出来る。

【００３７】ブラウザプログラムは、まず、ユーザから要求を受け取って、ユーザ名及びパスワードなどを検証して、そのユーザが特定の機能にアクセスすることが許可されているかを確かめるチェックを行う。

【００３８】次に、ブラウザプログラム１００は、適切なアルゴリズム及びジェネレータを利用してトークンを生成する。好ましい実施形態では、トークンは連続した数値ではないものであり、実際には乱数生成プログラムによって生成された数値である。

【００３９】そして、クライアントユーザインタフェースは、セキュリティサーバ1.４０に接続する。この接続は、例えば、ソケット（ socket ）を使用して行われる。ブラウザプログラム１００は、確立した接続、例えばソケット接続を利用して、セキュリティサーバ１４０にトークンを送信する。次いで、ブラウザプログラム１００は、サービスを要求するためにサーバアプリケーション１２０を呼び出し、要求されたサービスのための引数（ argument ）の一つとしてサーバアプリケーション１２０にトークンを送信する。このサービス要求は、ネットワーク回線を出てサーバシステム３１に送出され、サーバアプリケーション１２０に受信される。

【００４０】サーバアプリケーション１２０は、ブラウザプログラム１００からのサービス要求を受信する。次に、サーバアプリケーション１２０は、要求されたプログラムを調べ、サービスのためのプログラム名及び実行引数（ execute argument ）を用いて、ＣＧＩ－ＢＩＮプログラム１６０を呼び出す（ invoke）ことによって要求されたプログラムを呼び出す（ call）。

【００４１】ＣＧＩ－ＢＩＮプログラム１６０は、このプログラム名及び実行引数を受信する。要求されたサービスを提供する要求されたサブルーチンの実行前に、ＣＧＩ－ＢＩＮプログラム１６０が、セキュリティサーバ１４０とのソケットを確立する。一旦セキュリティサーバ１４０とのソケットが確立すると、ＣＧＩ－ＢＩＮプログラム１６０は、セキュリティサーバ１４０にトーク

9

ンを検証するためのトークン検証要求を送信する。

【００４２】セキュリティサーバ１４０は、初期化されると、受信ソケット（ listen socket）を確立する。セキュリティサーバ１４０は、ブラウザプログラム１００でユーザのアクセスが検証され許可されたときに生成された接続上に確立されているソケットでブラウザプログラム１００からのトークンを受信するのを待つ状態となる。トークンは、一旦クライアントユーザインタフェースから受信されると、セキュリティサーバ１４０のトークン検証テーブル（ token verification table ）に加えられる。セキュリティサーバ１４０は、ＣＧＩ－ＢＩＮプログラムとのトークンを検証するために確立されたソケットでＣＧＩ－ＢＩＮプログラム１６０からのトークン検証要求を受信するのを待つ状態となる。トークン検証要求がＣＧＩ－ＢＩＮプログラム１６０から受信されると、セキュリティサーバ１４０は、トークン検証テーブルをチェックし、そのトークンがクライアントユーザインタフェースから受信したものであり、したがって、正当なトークンであるか否かについて、ＣＧＩ－ＢＩＮプログラム１６０に対してトークン検証メッセージを返送する。

【００４３】ＣＧＩ－ＢＩＮプログラム１６０は、セキュリティサーバ１４０からトークン検証メッセージを受信すると、トークンの検証をチェックする。セキュリティサーバ１４０から受信したトークン検証メッセージが満足できるもの（ OK）である場合、すなわちトークンが正当なものである場合、ＣＧＩ－ＢＩＮプログラム１６０は、要求されたプログラムを実行し、標準出力（ std out ）に出力を書込む。次に、標準出力は、その出力をサーバアプリケーション１２０に返送する。セキュリティサーバ１４０から受信したトークン検証メッセージが許容できないものである場合、すなわちトークンが正当でない場合、エラーメッセージがサーバアプリケーション１２０に送信される。出力がサーバアプリケーション１２０に送信されると、ＣＧＩ－ＢＩＮプログラム１６０が終了して存在しなくなる。

【００４４】サーバアプリケーション１２０は、ＣＧＩ－ＢＩＮプログラム１６０の出力とＣＧＩ－ＢＩＮプログラム１６０のプロセスの終了ステータスとを受信し、ブラウザプログラム１００に対し、ネットワークを通じて出力を返送する。そして、ブラウザプログラム１００は、クライアントシステム１２においてサービスを要求したアプリケーションプログラムに対して出力を返送する。このプロセスについては、更に、図5から図9を参照して以下に説明する。

【００４５】図5は、図4に示すクライアントシステムのブラウザプログラムが実行するプロセスを示すフローチャートである。図4及び図5を参照して、ブラウザプログラム１００の最初のステップでは、ステップＳ１０１において、ブラウザプログラム１００が初期化され

10

る。そして、ブラウザプログラム１００は、ステップＳ１０２において、ユーザからユーザ名及びパスワードのログインを受け入れ、セキュリティサーバ１４０に接続する。次いで、ステップＳ１０３において、ユーザからサービス要求を受信する。

【００４６】次いで、ブラウザプログラム１００は、ステップＳ１０４においてトークンを生成する。好ましい実施形態では、トークンは乱数生成機能によって生成される乱数である。しかしながら、利用することが出来る一意的なトークンを生成する方法が他にもあることは、本技術分野において周知である。

【００４７】そして、ブラウザプログラム１００は、ステップＳ１０５において、ステップＳ１０４で生成されたトークンをセキュリティサーバ１４０に送信する。次いで、ステップＳ１０６において、サーバアプリケーション１２０に結合（ 又は接続）する。そして、ステップＳ１０７において、サーバアプリケーション１２０に対する呼び出しを行い、サーバアプリケーション１２０に対してトークンを引数のデータとして送信する。その後、ステップＳ１０８においてデータが返送されるまで処理を中断する。

【００４８】データがクライアントユーザインタフェースに戻ってくると、ブラウザプログラム１００は、ステップＳ１０８において再開し、ステップＳ１０９において、サーバアプリケーション１２０から受信した（ 出力された）データをユーザに対して表示する。そして、ステップＳ１１０のループを通じてステップＳ１０３に戻り、ユーザからの次のサービス要求を待つ状態になる。

【００４９】図6は、図4に示すサーバシステムのサーバアプリケーションが実行するプロセスを示すフローチャートである。サーバアプリケーション１２０は、ステップＳ１２１において初期化される。そして、ステップＳ１２２において、クライアントシステムからのサービス要求を受信するのを待つ状態となる。

【００５０】ステップＳ１２２においてサービス要求を受信すると、サーバアプリケーション１２０は、どのブラウザプログラム１００がクライアントシステムが要求したサービスを提供するかを判断し、ステップＳ１２３において、要求された特定のＣＧＩ－ＢＩＮプログラム１６０に結合する。サーバアプリケーション１２０は、ステップＳ１２４において、プログラム名及び実行引数等の特定の引数（ その中の一つはトークンである）で特定のＣＧＩ－ＢＩＮプログラムを呼び出し、必要なデータを送信する。サーバアプリケーション１２０は、ステップＳ１２５において、特定のＣＧＩ－ＢＩＮプログラム１６０からデータを受信するまで処理を中断する。

【００５１】サーバアプリケーション１２０は、ステップＳ１２６において、指定されたＣＧＩ－ＢＩＮプログラム１６０からの出力を受信する。そして、ステップＳ１２７において、ＣＧＩ－ＢＩＮプログラム１６０から

受信した出力を書き込み、またサービスを要求しているクライアントシステムに対してその出力を返送する。その後、ステップS128において、そのセッションを終了し、ステップS122に折返し、新たなサービス要求を受信するまで処理を中断する。

【0052】図7は、図4に示すサーバシステムのセキュリティサーバ140のプロセスを示すフローチャートである。まず、セキュリティサーバ140が、ステップS141において初期化される。次に、ステップS142において、ユーザのログイン名及びパスワードを得ることによって、ブラウザプログラム100からの接続（例えばソケット接続）を受け入れる。そして、ユーザのログイン名及びパスワードを認証する。一旦ログインユーザ名及びパスワードの認証が完了すると、セキュリティサーバ140は、ステップS143において、ソケット接続でクライアントユーザインタフェース100からのトークンを受信するまで処理を中断する。

【0053】セキュリティサーバ140は、ステップS144において、CGI－BINプログラム160からソケット接続を受け入れると、ステップS145において、CGI－BINプログラムとのソケット接続でCGI－BINプログラム160からのトークン検証要求を受信する。次いで、ステップS143においてクライアントユーザインタフェースからソケット接続で受信したトークンを用いて、CGI－BINプログラム160からCGI－BINプログラムのソケット接続で受信したトークンを検証する。

【0054】ステップS143において受信したトークンが、ステップS145において受信したトークンと一致する場合は、そのトークンの検証は合格となる。ステップS143においてクライアントユーザインタフェースからソケット接続で受信したトークンが、ステップS145においてCGI－BINプログラム160から受信したトークンと一致しない場合、そのークンの検証は不合格となる。セキュリティサーバ140は、所定の期間、タイムアウトする前にトークン検証要求がCGI－BINプログラム160から到着するのを待つ状態となる。トークンについて、タイムアウトしている場合にトークン検証要求がCGI－BINプログラム160から続けて送られてくると、トークンの検証が不合格となる。

【0055】セキュリティサーバ140は、ステップS147において、トークン検証の結果のトークン検証メッセージを、CGI－BINプログラム160に送信し、ステップS144において生成されたCGI－BINプログラムのソケット接続を閉じる。そして、ステップS148のループを通じてステップS143に戻り、クライアントユーザインタフェースから次のトークンを受信するまで待ち状態になる。

【0056】図8は、図4に示すCGI－BINプログラムプロセスのプロセスを示すフローチャートである。まず、CGI－BINプログラム160は、ステップS161において初期化される。そして、ステップS162において、プログラム名及び引数によって要求されたサービスに関するサービス要求を受信する。次いで、ステップS163において、セキュリティサーバ140に対するソケットを確立する。好ましい実施形態では、TCP／IPソケットが確立される。

【0057】ステップS164において、サーバアプリケーション120から受信したトークンを、検証のためにセキュリティサーバ140に送信する。そして、ステップS165において、セキュリティサーバ140からトークン検証メッセージが返送されるまで処理を中断する。

【0058】一旦セキュリティサーバ140からトークン検証メッセージを受信すると、ステップS166において、トークンの検証に関するテストを実行する。セキュリティサーバ140によってトークンが照合されると、プロセスはステップS167に進み、CGI－BINプログラム160が、要求されたサービスのプログラムを実行する。ステップS167において要求されたサービスのプログラムが実行された後に、ステップS168において、要求されたサービスのプログラムから標準出力及び標準エラーメッセージを受信する。CGI－BINプログラム160は、ステップS169において、サーバアプリケーション120に対して標準出力及び標準エラーデータを送信し、その後ステップS172において終了する。

【0059】ステップS166におけるトークンの検証をチェックした結果、トークンが照合さない場合、ステップS171において、CGI－BINプログラム160は、サーバアプリケーション120に対し、セキュリティサーバ140によるトークン検証が失敗したことを示すエラーメッセージを送信する。その後、CGI－BINプログラム160は、ステップS172において実行を終了する。

【0060】他の実施形態では、CGI－BINプログラム160が、セキュリティサーバ140に対し、実行中のコマンドのセキュリティレベルをトークンとともに送信する。セキュリティサーバ160は、トークンを検証するとともに、クライアントユーザインターフェース（ブラウザ）100のセキュリティレベルをチェックする。セキュリティサーバ140が正当なクライアントユーザインターフェース（ブラウザ）100をチェックしていることを保証するために、トークンを乱数でユーザインタフェースのセキュリティサーバ140に対する接続のポート数を加えたものとしても良い。クライアントユーザインターフェース（ブラウザ）100のセキュリティレベルは、セキュリティサーバ140が、セキュリティサーバ140への初期接続時においてクライアント

13

ユーザインターフェース(ブラウザ) １００を認証するときに決定される。

【００６１】前記した説明は、例示し、かつ、説明のために提示したものである。これは、全てを網羅することを意図したものではなく、本発明を開示したそのままの形態に限定することも意図していない。前記した教示内容を考慮して、明らかな変形例及び変更例が可能である。説明した実施の形態は、本発明の原理及びその実際的な適用例を最も良く例示するものとして選択し、その説明を行っており、これによって、当業者が、あらゆる実施の形態及び企図する特定の用途に適するようにあらゆる改良を加えることが可能である。すなわち、本発明を利用することが出来る。このような変形例及び変更例は全て、特許請求の範囲が公正にかつ合法的に権利を与えられる範囲に基づいて解釈されるときに、それら特許請求の範囲によって決定される発明の範囲内にある。

【００６２】以下に本発明の実施の形態を要約する。

１．通信ネットワークシステムにおけるウェブリソースの保護方法であって、クライアントブラウザ(１００) に対してトークンを生成するステップと、セキュリティサーバ(１４０) に対して前記トークンを転送するステップと、サービス要求及び前記トークンを転送するステップと、前記クライアントブラウザ(１００) からの前記サービス要求及び前記トークンを受信するアプリケーションサーバ(１６０) を提供するステップと、前記アプリケーションサーバ(１６０) によって前記トークンの検証を要求するステップと、前記アプリケーションサーバ(１６０) が受信した前記トークンを前記セキュリティサーバ(１４０) に転送するステップと、前記クライアントブラウザ(１００) から受信した前記トークンと前記アプリケーションサーバ(１６０) から受信した前記トークンとを比較するステップと、を有するウェブリソースの保護方法。

【００６３】２．前記クライアントブラウザ(１００) から受信したトークンと前記アプリケーションサーバ(１６０) から受信したトークンとが一致する場合に、一致通知を生成し、前記クライアントブラウザ(１００) から受信したトークンと前記アプリケーションサーバ(１６０) から受信したトークンとが一致しない場合に、不一致通知を生成するステップを更に有する上記1 記載のウェブリソースの保護方法。

【００６４】３．前記生成された通知を、前記アプリケーションサーバ(１６０) に転送するステップを更に有する上記2 記載のウェブリソースの保護方法。

【００６５】４．前記一致通知を受信したときに、前記要求されたサービスを提供するステップを更に有する上記3 記載のウェブリソースの保護方法。

【００６６】５．前記トークンを生成するステップに、乱数生成プログラムを利用して前記トークンを生成するステップを更に有する上記1 記載のウェブリソースの保護方法。

14

【００６７】６．通信ネットワークシステムにおいてウェブリソースを保護するコンピュータシステムにおいて、トークンを生成するクライアント装置(１２) と、サービスを提供するアプリケーション装置(１６０) と、前記クライアント装置で生成された前記トークンを検証するセキュリティ装置(１４０) とを備えるコンピュータシステム。

【００６８】７．前記トークンをサービス要求とともに前記アプリケーション装置(１６０) に転送する第1 のクライアント機構と、第三者による妥当性を検証するために前記トークンを前記セキュリティ装置のセキュリティサーバ(１４０) に転送する第2 のクライアント機構とを更に備える上記6 記載のコンピュータシステム。

【００６９】８．クライアントブラウザ(１００) からサービス要求及びトークンを受信する第1 のアプリケーション機構と、前記セキュリティ装置(１４０) に対して前記トークンの検証を要求する第2 のアプリケーション機構と、前記トークンの検証に関する一致通知を受信した際に、前記サービス要求で要求されたサービスを提供する第3 のアプリケーション機構とを更に備える上記6 記載のコンピュータシステム。

【００７０】９．クライアントブラウザ(１００) からトークンを受信する第1 のセキュリティ機構と、前記アプリケーション装置のアプリケーションサーバ(１６０) からのトークンを受信する第2 のセキュリティ機構と、前記クライアントブラウザ(１００) から受信したトークンと前記アプリケーションサーバ(１６０) から受信したトークンとを比較する第3 のセキュリティ機構と、を更に備える上記6 記載のコンピュータシステム。

【００７１】１０．前記クライアントブラウザ(１００) から受信したトークンと前記アプリケーションサーバ(１６０) から受信したトークンとが一致した際に、一致通知を生成し、かつ、一致しない場合に、不一致通知を生成する第4 のセキュリティ機構と、前記生成された通知を前記アプリケーションサーバ(１６０) に転送する第5 のセキュリティ機構と、を更に備える上記9 記載のコンピュータシステム。

【００７２】

【発明の効果】以上の説明から明らかなように、本発明のウェブリソースの保護方法によれば、クライアントシステムによって、サーバ上のウェブリソースにアクセスする際に、融通性のある、かつ高度なセキュリティの実施が出来るようになる。

【図面の簡単な説明】

【図1】本実施形態に係るインターネットを利用するクライアント／サーバシステムの一構成を示す構成図である。

【図2】実施形態にあってクライアントシステムのコンピュータシステム内のブラウザプログラムを示すブロッ

15

ク図である。

【図3】実施形態にあってサーバシステムのコンピュータシステム内のサービスアプリケーションプログラムとＣＧＩ－ＢＩＮプログラムとセキュリティサーバとを示すブロック図である。

【図4】クライアントユーザインターフェースと、サーバアプリケーションと、セキュリティサーバと、ＣＧＩ－ＢＩＮプログラムとのプロセスを示すブロック図である。

【図5】図4に示すクライアントシステムのブラウザプログラムが実行するプロセスを示すフローチャートである。

【図6】図4に示すサーバシステムのサーバアプリケーションが実行するプロセスを示すフローチャートである。

【図7】図4に示すサーバシステムのセキュリティサーバプログラムのプロセスを示すフローチャートである。

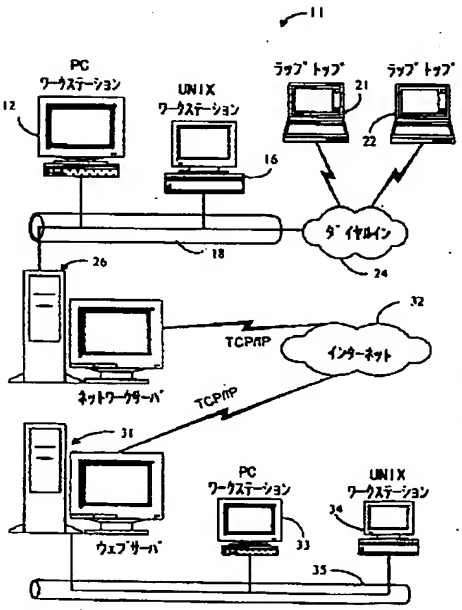
【図8】図4に示すＣＧＩ－ＢＩＮプログラムプロセスのプロセスを示すフローチャートである。

【符号の説明】

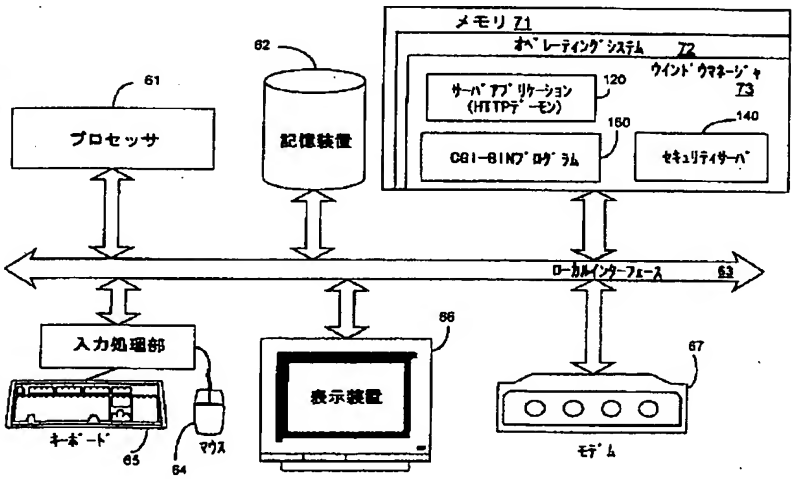
16

１１，１８　ネットワーク
１２，１６，２１，２２，３３，３４　ワークステーション（クライアントシステム）
２６　ネットワークサーバ（サーバシステム）
３１　ウェブサーバ（サーバシステム）
３２　インターネット
３５　ＬＡＮ
４１，６１　プロセッサ
４２，６２　記憶装置
４４，６４　マウス
４５，６５　キーボード
４６，６６　表示装置
４７，６７　モデム
５１，７１　メモリ
５２，７２　オペレーティングシステム
５３，７３　ウインドウマネージャ
１００　ブラウザプログラム
１２０　サーバアプリケーション
１４０　セキュリティサーバ
１６０　ＣＧＩ－ＢＩＮプログラム

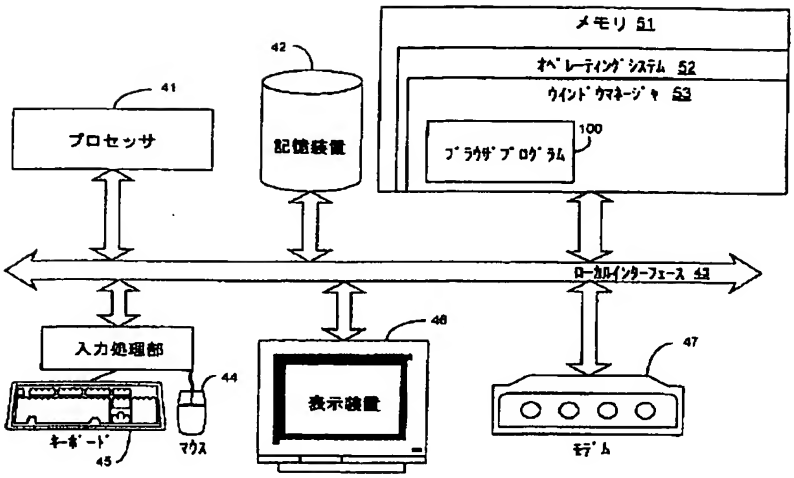
【図1】



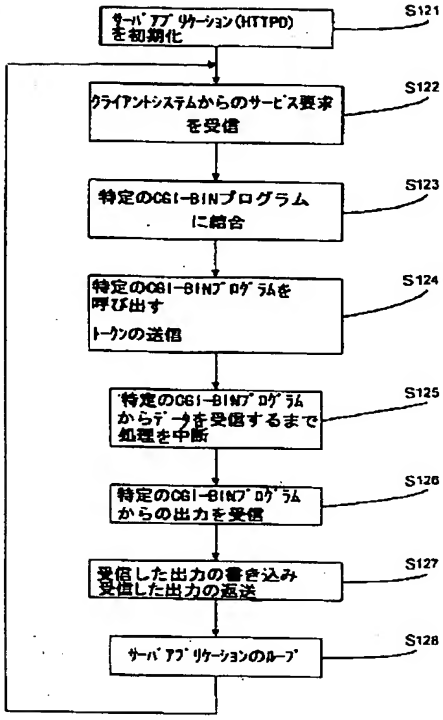
【図3】



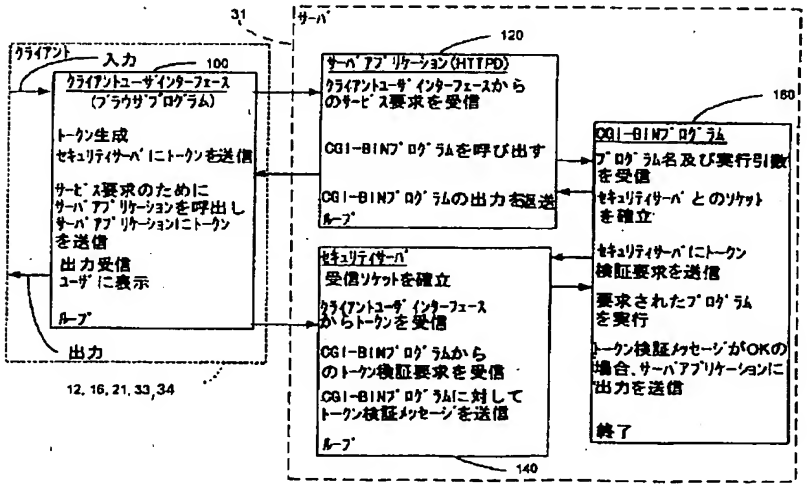
【図2】


メモリ 51
オペレーティングシステム 52
ウインドウマネージャ 53
ブラウザプログラム
100
42
41
プロセッサ
記憶装置
ローカルインターフェース 43
入力処理部
44
キーボード
45
マウス
46
表示装置
47
モデム

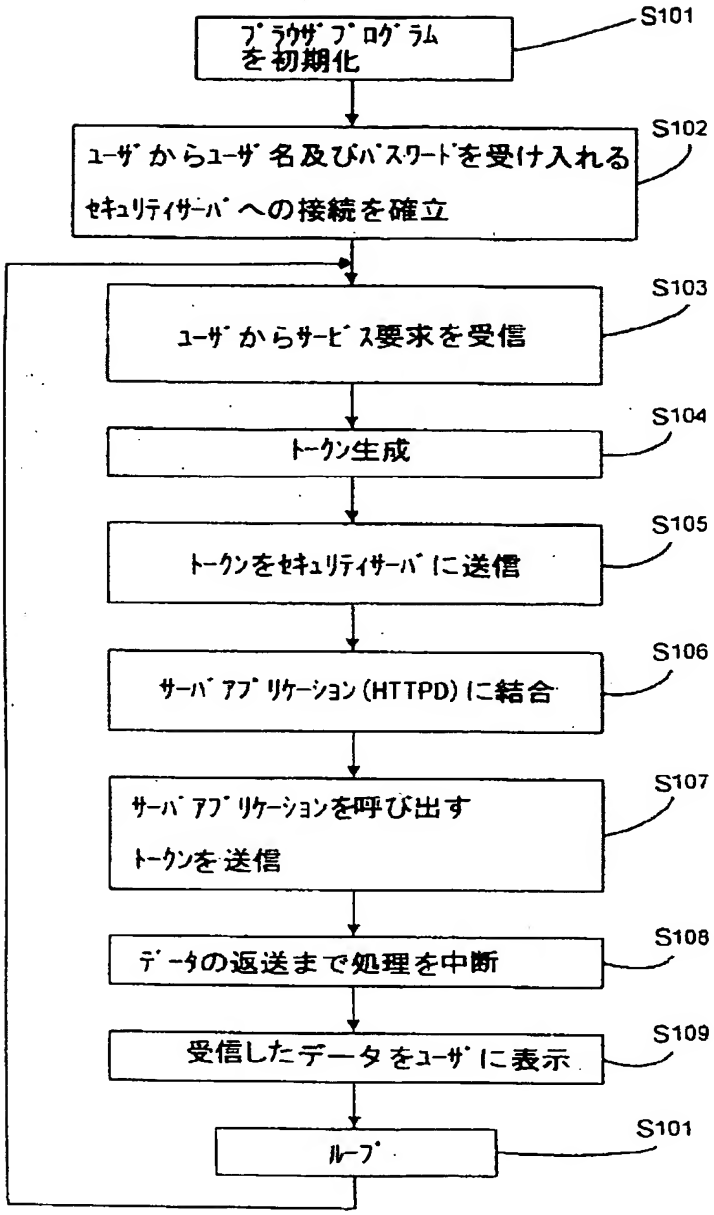
【図6】


サーバアプリケーション(HTTPD)を初期化
S121
クライアントシステムからのサービス要求を受信
S122
特定のCGI-BINプログラムに結合
S123
特定のCGI-BINプログラムを呼び出す
トークンの送信
S124
特定のCGI-BINプログラムからデータを受信するまで処理を中断
S125
特定のCGI-BINプログラムからの出力を受信
S126
受信した出力の書き込み
受信した出力の返送
S127
サーバアプリケーションのループ
S128

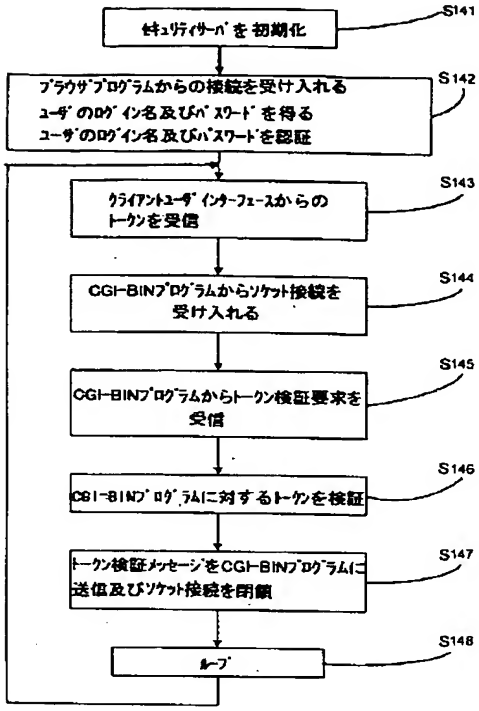
【図4】


31
クライアント
入力
100
クライアントユーザインターフェース
(ブラウザプログラム)
トークン生成
セキュリティサーバにトークンを送信
サービス要求のために
サーバアプリケーションを呼出し
サーバアプリケーションにトークンを送信
出力受信
ユーザに表示
ループ
出力
12, 16, 21, 33, 34
サーバ
120
サーバアプリケーション(HTTPD)
クライアントユーザインターフェースからのサービス要求を受信
CGI-BINプログラムを呼び出す
CGI-BINプログラムの出力を返送
ループ
セキュリティサーバ
受信ソケットを確立
クライアントユーザインターフェースからトークンを受信
CGI-BINプログラムからのトークン検証要求を受信
CGI-BINプログラムに対してトークン検証メッセージを送信
ループ
140
180
CGI-BINプログラム
プログラム名及び実行引数を受信
セキュリティサーバとのソケットを確立
セキュリティサーバにトークン検証要求を送信
要求されたプログラムを実行
トークン検証メッセージがOKの場合、サーバアプリケーションに出力を送信
終了

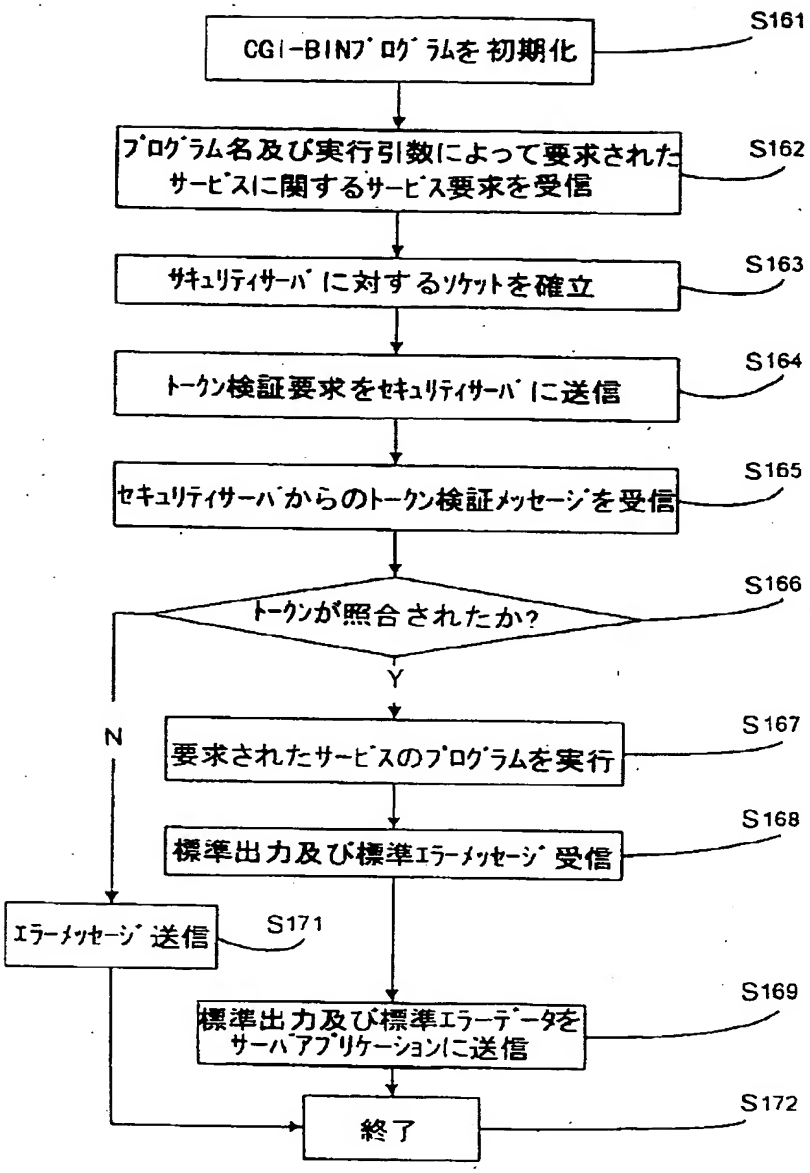
【図5】


ブラウザプログラムを初期化
S101
ユーザからユーザ名及びパスワードを受け入れるセキュリティサーバへの接続を確立
S102
ユーザからサービス要求を受信
S103
トークン生成
S104
トークンをセキュリティサーバに送信
S105
サーバアプリケーション(HTTPD)に結合
S106
サーバアプリケーションを呼び出すトークンを送信
S107
データの返送まで処理を中断
S108
受信したデータをユーザに表示
S109
ループ
S101

【図7】


セキュリティサーバを初期化
S141
ブラウザプログラムからの接続を受け入れる
ユーザのログイン名及びパスワードを得る
ユーザのログイン名及びパスワードを認証
S142
クライアントユーザインターフェースからのトークンを受信
S143
CGI-BINプログラムからソケット接続を受け入れる
S144
CGI-BINプログラムからトークン検証要求を受信
S145
CGI-BINプログラムに対するトークンを検証
S146
トークン検証メッセージをCGI-BINプログラムに送信及びソケット接続を閉鎖
S147
ループ
S148

【図8】



フロントページの続き

(72)発明者　トッド・エム・ゴイン
アメリカ合衆国　コロラド，ラブランド，ファイヤソーン　ドライブ　イースト　8320

(72)発明者　クライグ・ダブリュー・ブリアント
アメリカ合衆国　コロラド，フォートコリンス，ウエリントン・ドライブ　4321